(c) the restrictions to the march tests appear when we are trying to read or write to a specific memory address in the FIFO memory.

The other condition is reset where some tests like March C- doesn't work. For example March C- requires all the locations to be initialized to 0 in the start, but reset changes the values in the memory hence we cannot apply March C- test to FIFO.

Q4.

The paper "Memory Testing with a RISC Microcontroller" mainly discusses about testing low-end microcontroller which must apply low power on tests. They found out that the way to detect speed related faults is by performing memory tests at-speed, using Back-to-Back memory cycles which is minimum number of clock cycles for CPU. The paper uses ATMEL RISC microcontroller to demonstrate the capabilities and limitations of RISC CPU architectures.

It has 6 sections following the introduction which are in the following order:

2. ATMEL RISC microcontroller introduction
3. Explanation of the notation and specifying algorithms and shows test consequences of folding and scrambling.
4. Implementation of March elements.
5. A way to meet Back-to-Back cycle requirements.
6. Memory tests on ATMEL RISC CPU.
7. Conclusion.

Memory typically has a large number of words, while the number of bits per word is small. Folding is used such that the layout of the memory array approaches that of a square. It maps the logical memory view, consisting of words and bits/word, into the topological view, consisting of rows and columns. This paper assumes 4-way interleaved Folding.

      Address scrambling means that the Logical address sequence, as applied from the outside of the memory, differs from the physical internal address sequence. A very common cause for address scrambling is via sharing in the address decoders.

This paper considers MARCH C- test with following algorithm:
{↕(w0);⇑(r0,w1);⇑(r1,w0);⇓(r0,w1);⇓(r1,w0);↕(r0)}
If the architecture also allows access to smaller data types, then, for speed-related faults in the data multiplexers the following tests are applied for each of the smaller data types:
SCAN+ {⇑(w0);⇑(r0); ⇑(w1);⇑(r1);⇓(w0); ⇓(r0);⇓(w1);⇓(r1)}
The following are the assumptions and conventions made:
4-way interleaved folding (F=4). Products may even have 16-way folding; this is a simple extension of F=4. No address and/or data scrambling is assumed (to keep the algorithms more comprehensible).

There are two key issues for Back-to-Back cycle requirements. They are 1. Loop counting and 2. Read result evaluation for 'r0' operations.
    1. Loop counting:
        Consider the following:
        LDI R18, $M. Initialize Loop count
        CLR R31. Initialize Z-high
        LDI R30, $S. Initialize Z-low
        L1: ST Z+, R16. Perform a write operation

DEC R18. Decrement Loop count
BRNE L1. Branch if not equal
Here the problem is the memory was accessed only every 3 instructions in the loop which violates the Back-to-Back cycle requirement. So, the solution is Loop Unrolling which reduces the execution time by 58.3%. Similarly, paper addresses the case 2 problem also with example and solution which reduces the execution time by 37.5%.

Next the paper describes the capabilities and problems of ATMEL RISC architecture to implement memory test under Back-to-Back memory cycle requirements. It explains the testing in 2 sections first for data memory and then the program memory.
The following are the tests shown for data memory with examples:
1. MATS+ with linear Counting Method, Fast-Column and solid Data Background
2. MATS+ with linear Counting Method, Fast-Column and checkerboard Data Background
3. Address complement Counting Method
4. Fast-row Implementation.

Both the "MATS+ with linear Counting Method, Fast-Column and solid Data Background" and "MATS+ with linear Counting Method, Fast-Column and checkerboard Data Background" are explained with examples.
Address complement Counting Method is explained and proved with an example. It satisfies the Back-to-Back requirements using Z with post increment and Y with pre decrement Addressing Modes.
Fast-row addressing is not supported as it does not meet the Back-to-Back cycle requirement.
For program memory, both Fast-row and Address complement Counting Method cannot be applied, and it is concluded that Back-to-Back cycle requirements are not met.

Finally, the conclusion is that paper describes capabilities and faults of CPU-based memory testing. The tests in the paper are implemented in assemble language to satisfy Back-to-Back cycle requirements. It summarizes the data memory testing of ATMEL RISC CPU and then the program memory.

It concludes that tests must include the March C- to detest the static faults which are the state Coupling Faults (CFsts) and the static Address decoder Faults (AFs) can be detected with tests applied at any speed. it is recommended to provide more adequate architecture support for test implementation in next generation products. The results are a significant increase in fault coverage and a reduction in test time by about 60% when the test methods described in the paper are applied.